Description of Romberg:

It is one of numerical techniques that calculate integrals of functions, Romberg Integration based on successive application of the trapezoidal rule.

1. Trapezoidal rule:

It is one of the numerical rules to calculate the integration.

We need to calculate the approximate area under the carve, then we will divide this area into n trapezoidal, then we calculate the sum of trapezoidal area, we can easily calculate trapezoidal area use this formula:

$$I = (b-a)\frac{f(a)+f(b)}{2}$$

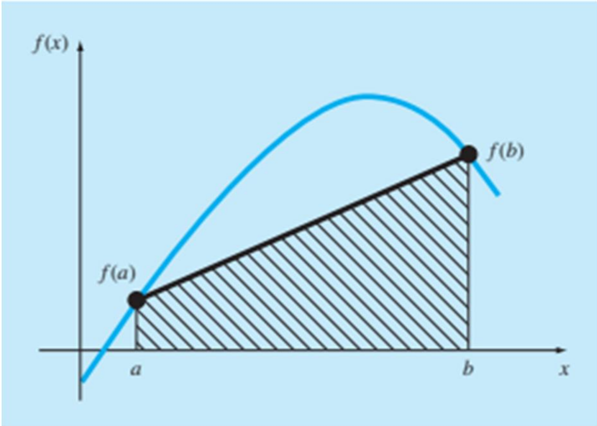

Figure 1 Graphical depiction of the trapezoidal rule.(Chapra, Canale and Jagadeeshan, 2015)

Then the general formula for n intervals is:

$$I = (b-a)\frac{f(Xo) + 2 * \sum_{i=1}^{n-1} f(Xi) + f(Xn)}{2n}$$

2. Richardson's Extrapolation:

these methods use two estimates of an integral to compute a third, more accurate approximation.

The estimate and error associated with a multiple-application trapezoidal rule can be represented generally as

$$I = I(h) + E(h)$$

where *I* = the exact value of the integral, *I(h)* = the approximation from an n-segment

application of the trapezoidal rule with step size h = (b - a)/n, and E(h) = the truncation error. If we make two separate estimates using step sizes of h1 and h2 and have exact values for the error,

$$I(h_1) + E(h_1) = I(h_2) + E(h_2)$$

the error of the multiple-application trapezoidal rule can be represented approximately by [with n= (b - a)/h]:

$$E \cong -\frac{b-a}{12} h^2 \overline{f''}$$

If it is assumed that $\overline{f''}$ is constant regardless of step size, previous equation can be used to determine that the ratio of the two errors will be

$$\frac{E(h1)}{E(h2)} \cong \frac{{h_1}^2}{{h_2}^2}$$

Then we can solve the previous 3 equations for

$$E(h2) \cong \frac{I(h_1) - I(h_2)}{I - ({}^{h_1}/_{h_2})^2}$$

Thus, we have developed an estimate of the truncation error in terms of the integral estimates and their step sizes. This estimate can then be substituted into

$$I = I(h) + E(h)$$

to yield an improved estimate of the integral:

$$I \cong I(h_2) + \frac{1}{\left({}^{h_1}/_{h_2}\right)^2 - 1}[I(h_2) - I(h_1)]$$

It can be shown (Ralston and Rabinowitz, 1978) that the error of this estimate is O(h^4). Thus, we have combined two trapezoidal rule estimates of O(h^2) to yield a new estimate of O(h^4). For the special case where the interval is halved (h2 = h1/2), this equation becomes

$$I \cong I(h_2) + \frac{1}{2^2 - 1}[I(h_2) - I(h_1)]$$

Or, collecting terms

$$I \cong \frac{4}{3}I(h_2) - \frac{1}{3}I(h_1)$$

3. The Romberg Integration Algorithm:

Then the general form for Romberg is

$$I_{j.k} \cong \frac{4^{k-1}I_{j+1.k-1} - I_{j.k-1}}{4^{k-1} - 1}$$

where$I_{j+1.k-1}$ and $I_{j.k-1}$ = the more and less accurate integrals, respectively, and $I_{j.k}$ = the improved integral. The index k signifies the level of the integration, where k = 1 corresponds to the original trapezoidal rule estimates, k = 2 corresponds to O(h^4), k = 3 to O(h^6), and so forth. The index j is used to distinguish between the more (j + 1) and the less (j) accurate estimates. For example, for k = 2 and j = 1, then the form becomes:

$$I_{1.2} \cong \frac{4I_{2.1} - I_{1.1}}{3}$$

Applications of Romberg in computer engineering:

- **Physics engines:**
  There are generally two classes of physics engines:
  High-precision physics engines require more processing power to calculate very precise physics and are usually used by scientists and computer animated movies.
  Real-time physics engines as used in video games and other forms of interactive computing use simplified calculations and decreased accuracy to compute in time for the game to respond at an appropriate rate for game play.
- **Modeling software**: like modeling for biological systems, meteorology and climatology, engineering applications, etc.
- **Graphing and visualization:**
  People use calculus in creating visuals or graphs. Often this visual/graph is 3D.
  They are used for video games especially physics engines.
- **Solve problems:**
  They are used for general problem solving application, simulation, and physics engine.
  Physics engine create realistic simulation in video games and probability simulations

And we need to calculate the integration in this equations. Romberg on of the numerical integration methods to calculate these integrations accurate.

References:

Chapra, S., Canale, R. and Jagadeeshan, R., 2015. *CHE3167*. North Ryde, N.S.W.: McGraw-Hill Australia, p.606, p.634:638.